

# On-line One Stroke Character Recognition Using Directional Features

Blind for review process<sup>1</sup>

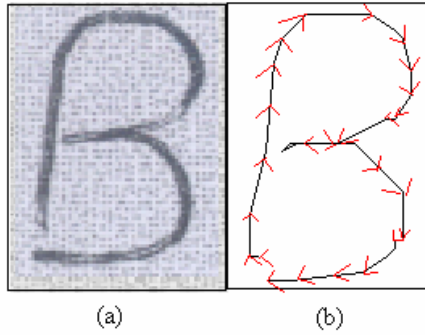
<sup>1</sup> Blind for review process

**Abstract.** This paper presents a method based on directional features for recognizing on-line characters represented by one-stroke. A new alphabet is proposed, and each character is represented as a vector of directional angles. This vector is then considered as a path in a decision tree in order to reach the character. The advantages of this method lie in its computational efficiency and robustness to noise.

## 1 Introduction

Handwriting recognition has many applications in different fields. These applications include reading hand-written notes on PDAs, bank cheques, postal addresses, and other fields on different forms. The persistent use of handwriting, in the era of digital computers, as the natural mean for storing data and information explains its convenience for numerous daily activities [1]. This extensive use of handwriting raises the importance of developing handwriting recognition systems.

When talking about handwriting recognition, one has to differentiate between two kinds of handwritten data: on-line and off-line. The former is obtained by writing with a special device such as a digital pen or on a special surface such as a PDA screen, while the latter is obtained through an image of an already written document, through scanning for example. In the on-line case, spatio-temporal information of the input data is available, while in the off-line case only spatio-luminance information of the image is available [1]. In other words, the on-line case provides the pen trajectory while the off-line case provides only the image of the written data; however, due to sampling, some of the information is lost. Figure 1 shows the difference between the two cases for the same character written with a digital pen. In Fig. 1.(a), the actual written pattern is shown, while Fig. 1.(b) shows the sampled version of this character which is reconstructed from the information obtained by the digital pen.



**Fig. 1.** (a) Off-line data (b) On-line data: Red arrows are added to show direction information

With the development of ink-aware modules, the concept of digital ink was introduced and a standard called Digital Ink Markup Language (InkML) was developed by the W3C as a format for representing ink that is entered by an input device [2]. Therefore, off-line processing of on-line data is now available. We believe that soon the distinction will no longer be on-line data and off-line data, but rather between on-line processing and off-line processing.

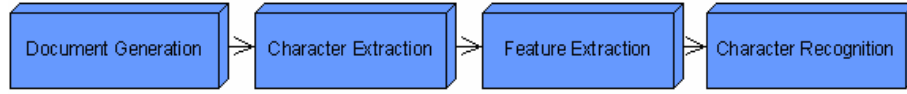
In this paper, we propose a method for on-line one stroke character recognition through directional features. The advantage of this method over previously existing ones [3,4] is its simplicity in extracting a compact robust feature for each character, in addition to the computational efficiency of recognition that does not depend on the concept of minimum distance, where each character has to be compared with all the alphabet.

This paper is organized as follows. Section 2 introduces the framework in which we developed our method. Section 3 presents the feature extraction method while section 4 presents the recognition method. In section 5, experimental results are shown. Concluding remarks are discussed in section 6.

## 2 Framework Overview

To obtain on-line information of the character strokes, a digital pen and a paper enabled with the Anoto functionality were used. The Anoto paper is a standard paper with a special pattern of dots printed on it [5]. These dots are invisible to the human eye and they are 0.3 mm apart. These dots are slightly shifted from a grid structure to form the proprietary Anoto pattern. A minute portion of the total pattern uniquely defines its position in the full pattern, 60,000,000 square kilometers, which is equivalent to an area exceeding that of Europe and Asia combined [5].

When writing with a digital pen on an Anoto paper, the movement of the pen is recorded through snapshots taken by a tiny camera inside the pen. The pattern of dots allows dynamic information coming from the digital camera in the pen to be processed into signals representing writing and drawing. This information is recorded on a memory chip found also inside the pen.



**Fig. 2.** System overview

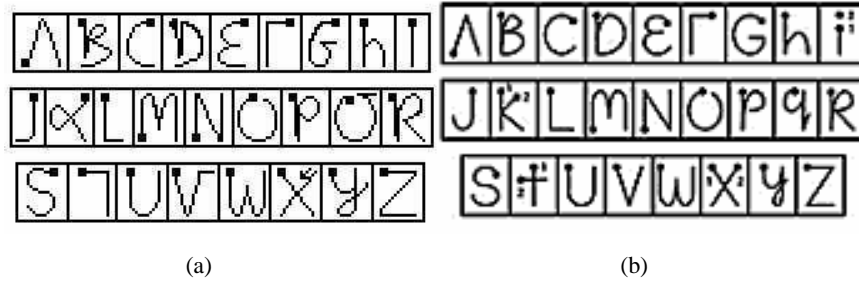
In Fig. 2, an overview of the system is shown. The document is generated from the information saved inside the digital pen. Extracting the characters from the document is simple, since we are assuming that the document contains only strokes that represent alphabet characters. Thereafter, the term character refers only to an alphabet character. The feature extraction step is discussed in section 3 while the character recognition step is discussed in section 4.

### 3 Features Extraction

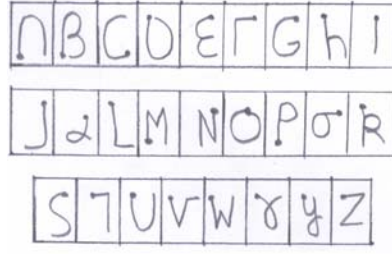
The feature used in recognizing different characters is the directional feature. This method tries to encode the pen trajectory into a vector of directional angles. The advantages of such a method are its robustness and its ability to extract general representative information from a given pattern.

#### 3.1 Alphabet

For representing each character in one stroke, a slightly modified version of the Graffiti Alphabet and the Graffiti 2 Alphabet, both designed by Palm Inc., is proposed, see Fig. 3. Our modification aims at a more robust representation by further differentiating the characters. For example, it can be seen, in Graffiti and Graffiti 2, that the stroke representations for B, P, and R are similar for a long period of the sketching time. In other words, to sketch B, P, or R one has to follow the same pen trajectory for a long time. Thus, our proposed alphabet is shown in Fig. 4.



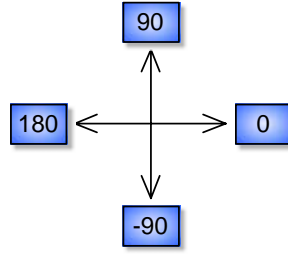
**Fig. 3.** (a) The Graffiti Alphabet by Palm, Inc. (b) The Graffiti 2 Alphabet by Palm, Inc.



**Fig. 4.** Our proposed Alphabet

### 3.2 Discrete Directional Features

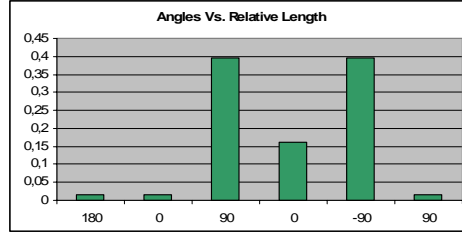
For recognition, each character is represented by a vector of normalized discrete directional angle values. These values are -90, 0, 90, and 180 as shown in Fig. 5. The angle value represents the normalized movement of the pen from the recorded points. Each character stroke is then represented by a vector of these values. For example, a stroke representing the character “A” will be recorded as the vector (90, 0, -90).



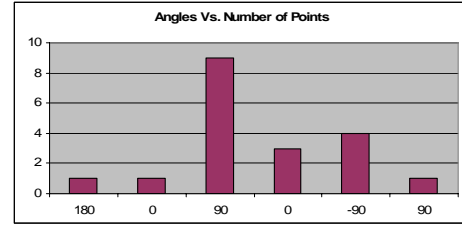
**Fig. 5.** Discrete normalized angle values

### 3.3 Filtering

Due to the way humans write, one expects noise while obtaining the angles vector. Also, since the normalization is in big proportions, more noise is introduced. Thus, filtering is necessary in order to obtain a representative vector that will be used in recognition. The filtering is based on the length of segment having the angular direction and the number of sampled points obtained from that direction. The idea here is that a sub-stroke (having a constant angular direction) is representative if it has a significant relative length or if the writer spent significant time writing it (a lot of sampled points were obtained), or both. For example, the angular vector obtained for one “A” stroke is (180, 0, 90, 0, -90, 90). The relative length of each sub-stroke having these angles is shown in Fig. 6, while the number of points forming each sub-stroke is shown in Fig. 7.



**Fig. 6.** Relative length of each sub-stroke



**Fig. 7.** Number of points for each sub-stroke

It can be easily seen from both Figures that the representative vector is (90, 0, -90). The algorithm used to generate the representative vector is the following:

*Given a vector of angles ( $a_1, a_2, \dots, a_n$ ) representing sub-strokes with the following set of relative lengths ( $r_1, r_2, \dots, r_n$ ) and the following number of points ( $p_1, p_2, \dots, p_n$ )*

*For every angle  $i$  in the vector*

*If  $r_i > T_r$  or  $p_i > T_p$*

*Keep  $a_i$  in the vector*

*Else if  $r_i > T'_r$  and  $p_i > T'_p$*

*Keep  $a_i$  in the vector*

*Else*

*Remove  $a_i$  from the vector*

where  $T_r$  is the threshold for the relative length and  $T_p$  is the threshold for the number of points.  $T'_r$  &  $T'_p$  are lower thresholds for the relative length and the number of points respectively.

## 4 Character Recognition

After the vector of representative angles is obtained, it is considered as a path in the decision tree to reach a character. The decision tree is shown in Fig. 8, and it is constructed by considering the vectors of angles that are used to form each character.

### 4.1 Multiple Paths

In most cases, there is more than one path to reach a character. For example, the path {90, 0, -90, 180} and the path {90, -90, 180} lead both to the character P. The purpose of having multiple paths is an attempt to cover all the possible generated vectors of representative angles in order to make the recognition more robust.

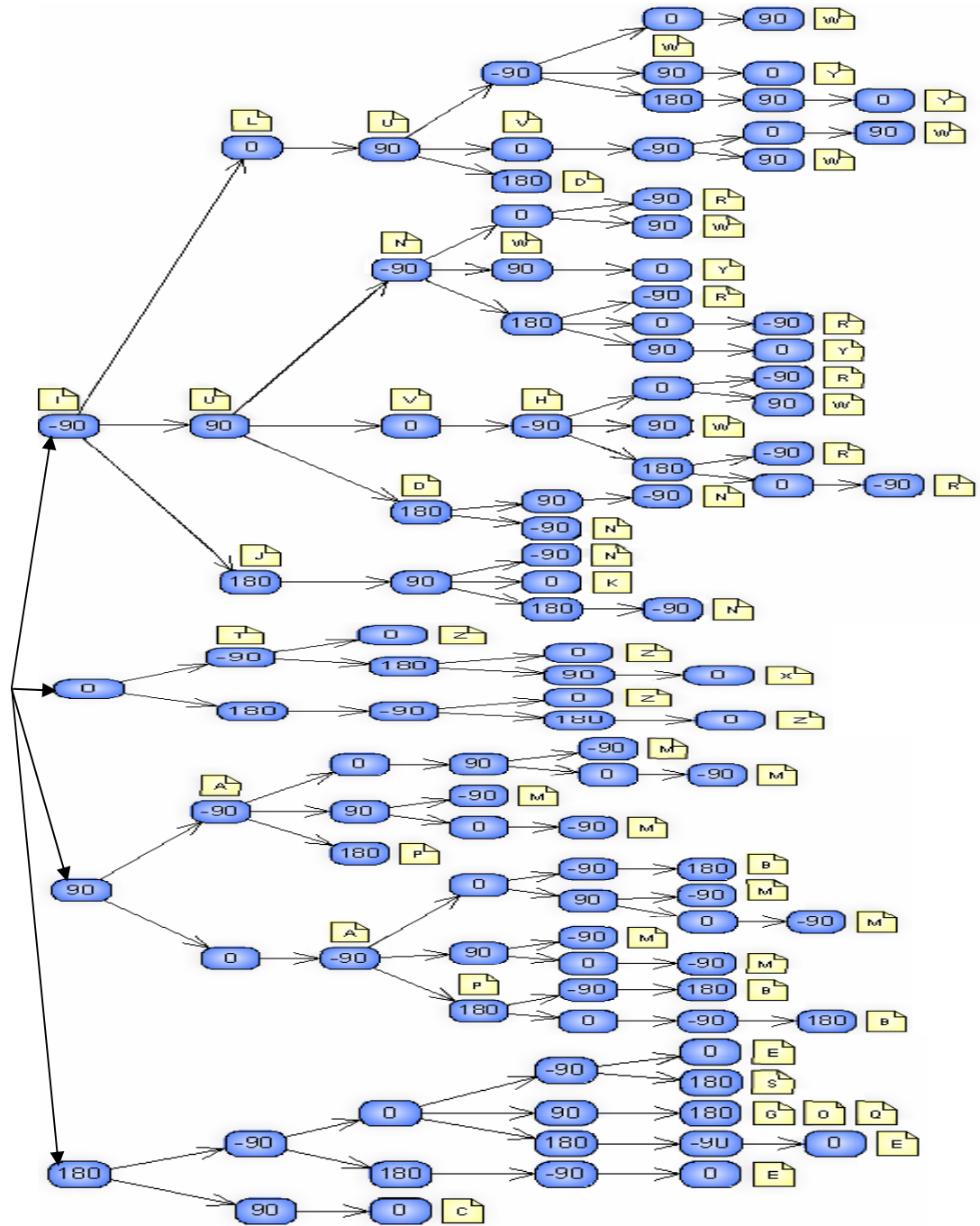


Fig. 8. Decision Tree

## 4.2 Robustness and Efficiency

It should be noted that the minimal and the most representative vector of angles is obtained; thus, the probability of this information being erroneous is extremely low.

The main advantage of this method is its computational efficiency: unlike other methods, it does not require comparing the obtained character with the rest of the alphabet for recognition.

## 4.3 Differentiating between G, O, and Q

The characters G, O, and Q have the same sequence of angles in the decision tree. However, noting that the initial sub-stroke (with  $180^\circ$  angle) in Q is longer than the rest and that the  $90^\circ$  stroke in G is shorter than that in O, the separation can be made.

## 5 Experimental Results

Our preliminary testing revealed some impressive results. With  $T_r$  set to 0.08,  $T_p$  set to 3,  $T'_r$  set to 0.05, and  $T'_p$  set to 2, we were able to achieve up to 100% recognition rate on a sample of 122 different characters. However, constructing a database with more samples, from different people, is compulsory to reach more conclusive results.

## 6 Conclusions

In this paper, we present an on-line one stroke character recognition method that is based on directional features. The advantages of this method are its computational efficiency and robustness to noise. The experimental results are promising. In the future, this method will be explored further and dynamic approaches to allocate the thresholds for obtaining the vector of angles will be considered.

## References

- [1] Plamondon, R. and Srihari, S. N., "On-line and Off-line Handwriting Recognition: A Comprehensive Survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no.1, pp. 63-84, January 2000.
- [2] W3C Multi-modal Interaction Working Group. Ink markup language (inkml). <http://www.w3.org/2002/mmi/ink>, 2007.
- [3] Long, T., Jin, L.W., Zen, L.X. and Huang, J.C., "One Stroke Cursive Character Recognition Using Combination of Directional and Positional Features", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 449-452, March 2005.
- [4] Ozer, O.F., Ozun, O., Tuzel, C.O., Atalay, V. and Cetin, A.E., "Vision-Based Single-Stroke Character Recognition for Wearable Computing", *IEEE Intelligent Systems*, vol. 16, issue 3, pp.33-37, May-June 2001.
- [5] Kolberg, M., Magil, E.H., Wilson, M., Birstwistle, P. and Ohlstenius, O., "Controlling Appliances with Pen and Paper", *Second IEEE Consumer Communications and Networking Conference*, pp. 156-160, January 2005.